



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/726,308	12/01/2003	Neal M. Gafter	SUN-P9044	3084

57960 7590 04/09/2007
SUN MICROSYSTEMS INC.
C/O PARK, VAUGHAN & FLEMING LLP
2820 FIFTH STREET
DAVIS, CA 95618-7759

EXAMINER

CHEN, QING

ART UNIT	PAPER NUMBER
----------	--------------

2191

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	04/09/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

Office Action Summary

Application No.

10/726,308

Applicant(s)

GAFTER ET AL.

Examiner

Qing Chen

Art Unit

2191

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 23 February 2007.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-3,5-10 and 12-14 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-3,5-10 and 12-14 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

1. This Office action is in response to the amendment filed on February 23, 2007.
2. **Claims 1-3, 5-10, and 12-14** are pending.
3. **Claims 1, 2, 5, 6, 8, 9, 12, and 13** have been amended.
4. **Claims 4, 11, and 15-21** have been cancelled.
5. The 35 U.S.C. § 112, second paragraph, rejections of Claims 1-3, 5-10, and 12-14 are withdrawn in view of Applicant's amendments to the claims.

Response to Amendment

Claim Rejections - 35 USC § 103

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

7. **Claims 1-3, 5, 6, 8-10, 12, and 13** are rejected under 35 U.S.C. 103(a) as being unpatentable over Thatte et al. (US 6,442,620) in view of Garcia et al. (US 6,778,990).

As per **Claim 1**, Thatte et al. disclose:

- receiving a first invocation of the software system (*see Column 9: 15-17, "... the computer 20 (FIG. 1) executes component applications that are developed as a package of component application objects."*; Column 10: 16-20, "A client requests instantiation of the COM

Art Unit: 2191

object using system-provided services and a set of standard, system-defined component interfaces based on class and interface identifiers assigned to the COM Object's class and interfaces. ");

- *assigning a first context to the first invocation (see Column 12: 8-14, "All objects 110-118 in the illustrated environment 100 (FIG. 3) are within a context, and have an association to the object context object 160 that represents the context. ");*

- *examining the first invocation to locate components of the first invocation (see Column 12: 40-46, "The "IObjectContext" interface 168 provides member functions to set or get the context property object for a given context property identifier, and to enumerate the context property objects of the object context object in order. The "IObjectContext" interface 168 also provides a member function to prevent further modification of its context properties (i.e., to freeze the context) after set-up. ");*

- *registering a unique factory to build each component, wherein the unique factory is registered using the first context (see Column 10: 24-31, "... classes of COM objects are uniquely associated with class identifiers ("CLSIDs"), and registered by their CLSID in a system configuration database referred to as the "registry." The registry entry for a COM object class associates the CLSID of the class with information identifying an executable file that provides the class (e.g., a DLL file having a class factory to produce an instance of the class). " and 41-47, "... the "CoCreateInstance()" API looks up the registry entry of the requested CLSID in the registry to identify the executable file for the class. The "CoCreateInstance()" API function then loads the class' executable file, and uses the class factory in the executable file to create an instance of the COM object 60. "); and*

- when a component is needed, building the component using the unique factory associated with the component, whereby building the component after each component has a registered factory eliminates potential problems with initialization circularity (*see Column 10: 44-49, "The "CoCreateInstance()" API function then loads the class' executable file, and uses the class factory in the executable file to create an instance of the COM object 60."*).

However, Thatte et al. do not disclose:

- a software design environment; and
- providing an additional factory for building an extended component of the first invocation.

Garcia et al. disclose:

- a software design environment (*see Column 1: 9-12, "The present invention relates generally to object-oriented development of database applications ..."*); and
- providing an additional factory for building an extended component of the first invocation (*see Column 3: 25-38, "The CAM is a singleton object that maps unique component ID's to a Factory Creation Function (FCF) in a dynamic library. A purpose of the FCF is to create factories and register them with a database connection."* and 45-53, *"When the application needs to extend functionality during run-time operation, the method of dynamic component activation is used."* and *"The object factory can be used to create the objects that provide the additional functionality. The database connection object receives this request and checks in a local lookup table containing a map between object ID's and the Factory Creation Functions (FCF)s."*).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Garcia et al. into the teaching of Thatte et al. to include a software design environment; and providing an additional factory for building an extended component of the first invocation. The modification would be obvious because one of ordinary skill in the art would be motivated to create software that is extensible while running in a heterogeneous and distributed environment (see Garcia et al. – Column 3: 4-9).

As per **Claim 2**, the rejection of **Claim 1** is incorporated; and Thatte et al. further disclose:

- receiving a second invocation of the software system (see Column 9: 15-17, "... the computer 20 (FIG. 1) executes component applications that are developed as a package of component application objects."; Column 10: 16-20, "A client requests instantiation of the COM object using system-provided services and a set of standard, system-defined component interfaces based on class and interface identifiers assigned to the COM Object's class and interfaces.");
- assigning a second context to the second invocation (see Column 12: 8-14, "All objects 110-118 in the illustrated environment 100 (FIG. 3) are within a context, and have an association to the object context object 160 that represents the context.");
- examining the second invocation to locate components of the second invocation (see Column 12: 40-46, "The 'IObjectContext' interface 168 provides member functions to set or get the context property object for a given context property identifier, and to enumerate the context property objects of the object context object in order. The 'IObjectContext' interface 168 also

Art Unit: 2191

provides a member function to prevent further modification of its context properties (i.e., to freeze the context) after set-up. ");

- registering a unique factory to build each component in the software design environment, wherein the unique factory is registered using the second context (see Column 10: 24-31, "... classes of COM objects are uniquely associated with class identifiers ("CLSIDs"), and registered by their CLSID in a system configuration database referred to as the "registry." The registry entry for a COM object class associates the CLSID of the class with information identifying an executable file that provides the class (e.g., a DLL file having a class factory to produce an instance of the class)." and 41-47, "... the "CoCreateInstance()" API looks up the registry entry of the requested CLSID in the registry to identify the executable file for the class. The "CoCreateInstance()" API function then loads the class' executable file, and uses the class factory in the executable file to create an instance of the COM object 60. "); and

- when a component is needed, building the component using a factory associated with the component, whereby building the component after each component has a registered factory eliminates problems with initialization circularity (see Column 10: 44-49, "The "CoCreateInstance()" API function then loads the class' executable file, and uses the class factory in the executable file to create an instance of the COM object 60. ").

As per **Claim 3**, the rejection of **Claim 2** is incorporated; and Thatte et al. further disclose:

- wherein components from the second invocation are not available to the first invocation (see Column 13: 31-33, "In COM+, a reference to an object in another context

Art Unit: 2191

(herein termed a "cross-context object reference" 182-183) is indirect via an object context switcher." and 53-58, *"In the case where the client and server component application objects 190, 192 are in different apartments or processes (i.e., a cross-apartment or process, cross-context reference 183), the object context switcher is implemented as a proxy 186 and stub 187 pair, which performs marshaling as well as context switching.").*

As per **Claim 5**, the rejection of **Claim 1** is incorporated; and Thatte et al. further disclose:

- wherein registering the unique factory to build each component in the software design environment involves placing a key and a related factory identifier into a storage structure *(see Column 10: 24-27, "... classes of COM objects are uniquely associated with class identifiers ("CLSIDs"), and registered by their CLSID in a system configuration database referred to as the "registry."")*.

As per **Claim 6**, the rejection of **Claim 5** is incorporated; and Thatte et al. further disclose:

- wherein building the component in the software design environment using the unique factory associated with the component involves using the key to retrieve the related factory identifier from the storage structure *(see Column 10: 27-31, "The registry entry for a COM object class associates the CLSID of the class with information identifying an executable file that provides the class (e.g., a DLL file having a class factory to produce an instance of the class).")*.

Claims 8-10, 12, and 13 are computer-readable storage device claims corresponding to the method claims above (Claims 1-3, 5, and 6) and, therefore, are rejected for the same reasons set forth in the rejections of Claims 1-3, 5, and 6.

8. **Claims 7 and 14** are rejected under 35 U.S.C. 103(a) as being unpatentable over Thatte et al. (US 6,442,620) in view of Garcia et al. (US 6,778,990) as applied to Claims 6 and 13 above, and further in view of Gibbons et al. (US 5,761,511).

As per **Claim 7**, the rejection of **Claim 6** is incorporated; however, Thatte et al. and Garcia et al. do not disclose:

- wherein the storage structure includes a hash table.

Gibbons et al. disclose:

- wherein the storage structure includes a hash table (*see Column 9: 33-35, "... implementing a shadow map by calling upon a factory object and caching the results in a hash table ..."*).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Gibbons et al. into the teaching of Thatte et al. to include wherein the storage structure includes a hash table. The modification would be obvious because one of ordinary skill in the art would be motivated to provide a constant lookup time on average, regardless of the number of items in the hash table.

Claim 14 is rejected for the same reason set forth in the rejection of Claim 7.

Response to Arguments

9. Applicant's arguments with respect to Claims 1, 2, 5, 6, 8, 9, and 12 have been considered, but are moot in view of the new ground(s) of rejection.

Conclusion

10. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire **THREE MONTHS** from the mailing date of this action. In the event a first reply is filed within **TWO MONTHS** of the mailing date of this final action and the advisory action is not mailed until after the end of the **THREE-MONTH** shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than **SIX MONTHS** from the date of this final action.

Any inquiry concerning this communication or earlier communications from the Examiner should be directed to Qing Chen whose telephone number is 571-270-1071. The Examiner can normally be reached on Monday through Thursday from 7:30 AM to 4:00 PM. The Examiner can also be reached on alternate Fridays.

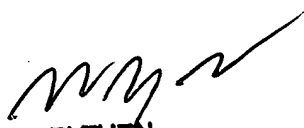
Art Unit: 2191

If attempts to reach the Examiner by telephone are unsuccessful, the Examiner's supervisor, Wei Zhen, can be reached on 571-272-3708. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the TC 2100 Group receptionist whose telephone number is 571-272-2100.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

QC / QC
March 14, 2007


WEI ZHEN
SUPERVISORY PATENT EXAMINER